

2/PR45

526 Rec'd PCT/PTO 02 MAY 2000

1

Beschreibung

Verfahren und Vorrichtung zur Systemsimulation von Mikrocontrollern/Mikroprozessoren und zugehörnden Peripheriemodulen

5

Die Erfindung betrifft ein Verfahren zur Systemsimulation sowie eine Vorrichtung zur Durchführung des Verfahrens.

Simulationen von Rechnermodulen dienen dazu, die Systemarchitektur eines Rechners zu optimieren, eine geeignete Software zu entwickeln und frühzeitig Fehler zu erkennen, bevor die Hardware tatsächlich realisiert wird. Dadurch kann erheblicher Entwicklungsaufwand eingespart werden.

Muster eines Systems ist hierbei die Anordnung eines Microcontrollers mit Peripheriemodulen und externer Umgebung zu verstehen. Der Microcontroller im speziellen ist die reine Programmverarbeitungseinheit (CPU). Im allgemeinen besitzen Microcontrollerchips aber auch Peripheriemodule auf dem Chip. Peripheriemodule können hierbei die unterschiedlichsten Aufgaben besitzen, z. B. Analog zu Digital Wandler (ADC), Zähler, serielle Schnittstellen usw. Peripheriemodule sind in diesem Zusammenhang immer Hardwaremodule.

Wenn zum Beispiel für einen Mikrocontroller ein Peripheriemodul entwickelt werden soll, wird dieses Modul mit typischen Signalmustern simuliert, wobei der vorzugsweise ebenfalls simulierte Mikroprozessor das Modul konfiguriert und steuert. Die simulierten Zustände des Moduls werden durch den Mikroprozessor abgefragt und ausgewertet. Da die Simulationen taktzyklengenau sind, werden alle Module immer mitsimuliert, wenn der Mikrocontroller die Zustände des Peripheriemoduls auswertet. Diese Auswertung kann die Simulationszeit erheblich verlängern.

35

Insbesondere in dem Fall, in dem mehrere Module gleichzeitig simuliert werden, kann jedoch der gesamte Ablauf eine relativ lange Zeitdauer erfordern, zumal aufgrund des einheitlichen Prozessortaktes die Simulation aller Module stets weiterläuft, auch wenn gerade die Zustände eines Moduls ausgewertet werden.

Bisher wurde die verlängerte Simulationszeit in Kauf genommen oder es wurde die Auswertung der Peripheriezustände auf ein Minimum reduziert bzw. nicht während der Systemsimulation gemacht.

Aus EP 0 777 180 A2 ist bereits ein Verfahren zur Simulation und Emulation von Systemen, bestehend aus Software und (simulierten) Hardware Komponenten, bekannt. Das Verfahren sieht vor, die Interaktion zwischen den Hardware und Software Komponenten zu ermöglichen und gleichzeitig beide Systeme weitgehend zu entkoppeln.

Die Grundidee dieses Verfahrens ist es, die Uhren der Komponenten unabhängig laufen zu lassen und sie nur an den Stellen, an denen eine Interaktion zwischen den Systemen erfolgt, für die unbedingt notwendige Zahl von Taktzyklen zu synchronisieren. (Anm.: Hierbei ist der Ausdruck "Uhr einer Komponente" in dem Sinne verwendet, daß damit die simulierte Zeit gemessen wird.) Die reale Zeit ist natürlich für alle Komponentensimulationen gleich. Um den Unterschied zu verdeutlichen: Die eine Sekunde simulierte Zeit kann z. B. eine Stunde reale Zeit dauern.

Der Vorteil dieses Verfahrens liegt darin, daß eine schnell simulierbare Systemkomponente nicht von einer langsamen ausgebremst wird. Das gilt natürlich nur, wenn die Interaktion zwischen den Komponenten gering ist. Sind dagegen beide Uhren starr synchronisiert, dann kann die Simulationsgeschwindigkeit niemals höher als die der langsamsten Komponenten sein.

Der Nachteil des Verfahrens liegt darin, daß es in vielen Systemsimulationen nicht tolerierbar ist, daß die Uhren der Komponenten keinerlei absolute Synchronisation besitzen. Hinzu kommt, daß Simulationsläufe nicht völlig reproduzierbar sind, da die relative Stellung der Uhren der Teilkomponenten z. B. durch die Auslastung der Simulationsrechner beeinflußt wird.

Der Erfindung liegt deshalb die Aufgabe zugrunde, ein Verfahren und eine Vorrichtung der eingangs genannten Art anzugeben, mit dem der gesamte Simulationsablauf wesentlich beschleunigt werden kann.

Gelöst wird diese Aufgabe für das Verfahren dadurch, daß eine erste Abfolge von Verfahrensschritten zur Simulation des Moduls mit vorbestimmten Signalmustern und eine zweite Abfolge von Verfahrensschritten zur Abfrage und zur Auswertung von durch die Simulation hervorgerufenen Systemzuständen vorgesehen ist, wobei die erste Abfolge zur Durchführung der zweiten Abfolge nach Maßgabe von in die erste Abfolge eingefügte Marken unterbrochen und die zweite Abfolge in einer an die Auswertung angepaßten, beschleunigten Betriebsart ausgeführt wird.

Während der normalen Simulation wird der Mikrocontroller und das Peripheriemodul taktzyklengenau simuliert, während bei der beschleunigten Codeausführung dagegen keine "simulierte" Zeit vergeht, d. h., daß der Programmteil in einer Art Instruction-Set-Simulator abgearbeitet wird.

Unter beschleunigter Betriebsart bzw. Codeausführung ist dabei zu verstehen, daß nur ein geringer Teil des Systems imuliert/ausgeführt wird und das zum Teil noch in einer vereinfachten Form. Ein Beispiel ist, daß nur die CPU als Befehlssatz-Simulator Programmcode verarbeitet und das übrige System nicht simuliert wird. Ein Beispiel für ein Peripheriemodul

ist eine serielle Schnittstelle, die Daten im Ausgangspuffer direkt im Speicher für die Simulationsauswertung ablegt. Im Falle der nichtbeschleunigten Betriebsweise, würde die serielle Schnittstelle die Daten bitweise über mehrere Taktzyklen hinweg in ein Ausgangssignal umsetzen, das dann von einem Empfänger bitweise empfangen, zusammengesetzt und im Speicher abgelegt wird.

In einer Ausführungsform ist vorgesehen, daß während der beschleunigten Codeausführung auch bestimmte Peripheriemodule rein funktional mitsimuliert werden.

Die Aufgabe wird für die Vorrichtung zur Durchführung des Verfahrens dadurch gelöst, daß eine Mikroprozessor-Steuereinheit zur Simulation des Moduls vorgesehen ist durch Erzeugung von taktzyklenbasierten Signalmustern sowie zur Abfrage und zur Auswertung der durch die Simulation hervorgerufenen Systemzustände während einer Programmunterbrechung durch Aktivierung eines Befehlssatz-Simulators.

Ein besonderer Vorteil dieser Lösungen besteht darin, daß durch die Trennung der eigentlichen Systemsimulation von der Auswertung der Simulationsergebnisse eine wesentliche Verringerung der für den Gesamtablauf erforderlichen Zeit erreicht werden kann, da in der Auswertungsphase der Prozessor einerseits nicht mit den dann überflüssigen Simulationsvorgängen belastet ist und andererseits die Auswertung an sich durch die beschleunigte Betriebsart schneller abläuft.

Weitere Vorteile bestehen darin, daß das Programm die Zustände des Peripheriemoduls umfassend kontrollieren und testen kann. Es muß nicht ein zusätzliches externes Auswertungsprogramm aktuell gehalten werden. Darüber hinaus kann vollständig verhindert werden, daß die Auswertung das Zeitverhalten des Programms beeinflusst.

Ein entscheidender Vorteil besteht darüber hinaus darin, daß das für die Systemsimulation und den simulierten Mikrocontroller erarbeitete Programm nach Entfernung der vorgesehenen Marken für den tatsächlich realisierten, also in Silizium gegossenen Mikrocontroller eingesetzt werden kann.

Zusammenfassend besteht die Erfindung zur Simulation von Rechnermodulen im Wesentlichen darin, daß sich die Systemsimulation aus der Sicht des Mikrocontrollers bzw. Mikroprozessors in zwei Teilsimulationen unterteilen lassen: Zum einen in die eigentliche Systemsimulation, also in die Simulation des Peripheriemoduls, das mit typischen Signalmustern simuliert wird, und in die Simulation des Mikrocontrollers. Die zweite Teilsimulation betrifft die Auswertung von abgefragten Systemzuständen. Wenn die Auswertungsphase durch das vorgeschlagene Verfahren beschleunigt wird, wird sowohl das simulierte Zeitverhalten genauer, als auch die gesamte Simulationszeit wesentlich verringert.

Das erfindungsgemäße Verfahren wird nachfolgend anhand eines Ausführungsbeispiels näher erläutert.

Als Ausführungsbeispiel wird die Simulation eines Mikrocontrollers vom Typ 8051 angenommen.

Beim Simulationsmodell des 8051-Mikrocontrollers sei angenommen, daß das erfindungsgemäße Verfahren implementiert ist. Beispielsweise kann der Assemblercode für das Simulationsmodell des 8051-Mikrocontrollers folgendermaßen lauten:

```
... (Programcode) ...
```

```
db 0a5h, "l+" ; **** start lightspeed mode
```

```
mov sbuf, #"H" ; visible in the console window
```

```
mov sbuf, #"i"
```

```
mov sbuf, #"!"
```

```

db 0a5h, "1-"      ; **** end lightspeed mode

mov sbuf, #"H"      ; visible on the internal bus, but
5                      not in the console window
                      ... (Programmcode) ...

```

Der Programmcode wird zwischen den Marken direkt abgearbeitet, ohne daß das Simulationsmodell Taktflanken benötigt. Der hierbei abgearbeitete Programmcode befindet sich vor den Marken und nach den Marken und ist oben lediglich durch Punkte (...) angedeutet. Als Marken können die üblicherweise nicht verwendeten Opcode a5h mit der nachfolgenden ASCII-Zeichenfolge "1+" und "1-" zum Starten bzw. zum Beenden der zweiten Abfolge von Verfahrensschritten verwendet werden. Auch geeignete Opcodesequenzen können verwendet werden. Die zweite Abfolge von Verfahrensschritten wird nachfolgend als "lightspeedmode" bezeichnet.

Während dieser zweiten Abfolge von Verfahrensschritten wird die serielle Schnittstelle des Simulationsmodells des 8051-Mikrocontrollers nachgebildet, indem alle Ausgaben vom Simulationsmodell auf ihr Register SBUF direkt an das Konsolenfenster geschrieben werden.

Anhand eines weiteren Ausführungsbeispiels wird die Erfindung im Zusammenhang von Figuren weiter erläutert. Es zeigen:

Fig. 1: ein Blockschaltbild für eine Anordnung zur Gesamtsystemsimulation nach der Erfindung, und

Fig. 2: einen Ausschnitt von Verfahrensschritten einer CPU, die nach dem erfindungsgemäßen Verfahren betrieben wird.

In Fig. 1 ist ein Blockschaltbild für eine Anordnung zur Gesamtsystemsimulation nach der Erfindung dargestellt. Das Blockschaltbild zeigt den Kern eines Microprozessors μC , der eine CPU-Einheit CPU, einen Programmspeicher PS und einen Datenspeicher DS enthält, der Microprozessor μC ist mit einer Peripherieeinheit P in Verbindung, die mehrere Peripheriemodule PM1, PM2, PM3 bis Pmn aufweist. Zusätzlich sind zwei Blöcke für die simulierte Umgebung des Peripheriemoduls PM1 und des Peripheriemoduls PM3 dargestellt. Die Blöcke sind mit den Bezugszeichen SPM1 und SPM3 bezeichnet. Der Block SPM1 könnte beispielsweise ein simulierter Sinusgenerator sein. Der mit dem Bezugszeichen SPM3 bezeichnete Block könnte beispielsweise eine simulierte Konsole sein. Als Peripheriemodul PM1 kann beispielsweise ein Analog-/Digital-Wandler als Peripheriemodul PM2 ein Zähler und als Peripheriemodul PM3 eine serielle Schnittstelle vorgesehen sein. Sämtliche Komponenten, also der Microprozessor μC , die Peripheriemodule PM1 bis Pmn und die simulierten Umgebungen für die Peripheriemodule stehen mit einer gemeinsamen Zeitbasis clk, also einem gemeinsamen Takt, miteinander in Verbindung.

Eine Systemsimulation könnte bei einer solchen Anordnung beispielsweise folgendermaßen aussehen. Das Peripheriemodul PM1, also z. B. Analog-/Digital-Wandler, mißt die Spannung des simulierten Sinusgenerators und löst nach jeder Messung einen Interrupt aus. Die CPU liest daraufhin den Wert aus dem Ergebnisregister des Analog-/Digital-Wandlers und schreibt ihn in den Datenspeicher DS. Nach einer bestimmten Anzahl von Messungen, schaltet sich die CPU in die beschleunigte Betriebsart und wertet diese Messungen aus. Nachdem sie dies gemacht hat, schaltet sie in den normalen Moduls zurück und die Simulation läuft genau an der Stelle weiter, an der umgeschaltet wurde. Darum ist die Systemsimulation völlig unbeeinflusst von der Auswertung.

In Fig. 2 ist getragen auf simulierte Zeittakte 0 bis 6 der Status des Analog-/Digital-Wandlers (ADC) und der CPU bzw. deren Befehle dargestellt. Die CPU startet demnach eine ADC-Wandlung und kopiert das Ergebnis in den Speicher. Im

- 5 Lightspeed-Modus zur Testauswertung wird dann getestet, ob der Erwartungswert 16 ± 1 gemessen wurde. Dieser Test ist ohne Einfluß auf die Gesamtsystemsimulation, da die Wandlung 2 ohne Verzögerung gestartet wird.
- 10 Bei dem erfindungsgemäßen Verfahren sind die "Uhren aller Teilkomponenten" grundsätzlich starr gekoppelt und laufen synchron. Die Ausnahme bildet lediglich der sogenannte Lightspeed-Modus, bei dem die Uhr der Hardware Komponenten stillsteht und die Software auf der CPU jedoch abläuft. Vor
- 15 einem Zugriff von der Software auf die Hardware muss der Lightspeed-Modus explizit verlassen werden, was durch spezielle Marken, die ansonsten im Programm nicht vorhanden sind, ausgelöst wird.
- 20 Der Grund für die zwei verschiedenen Modi bei dem erfindungsgemäßen Verfahren liegt in der Möglichkeit der vollständigen Kontrolle der Uhr aller Teilkomponenten. Dies kann vorteilhafterweise für zwei Anwendungen genutzt werden:
- 25 1. Auf der simulierten CPU können Testprogramme ausgeführt werden, ohne daß die simulierte Zeit vergeht und damit die Systemsimulation an sich beeinflusst wird.
- 30 2. Die Simulation kann beschleunigt werden, wenn die Software nur auf der CPU läuft, ohne daß die Uhr der Hardware Komponenten weiterläuft.

Im Gegensatz zu dem bekannten Verfahren ist das erfindungsgemäße Verfahren deterministisch und reproduzierbar.

Patentansprüche

1. Verfahren zur Systemsimulation mit simulierten Mikrocontrollern/Mikroprozessoren und zugehörender Peripheriemodulen,
5 g e k e n n z e i c h n e t d u r c h
eine erste Abfolge von Verfahrensschritten zur Simulation des Mikrocontrollers/Mikroprozessors und der Peripheriemodule mit vorbestimmten Signalmustern und
eine zweite Abfolge von Verfahrensschritten zur Abfrage und
10 zur Auswertung von durch die Simulation hervorgerufenen Systemzuständen,
wobei die erste Abfolge zur Durchführung der zweiten Abfolge nach Maßgabe von in die erste Abfolge eingefügte Marken unterbrochen und die zweite Abfolge in einer an die Auswertung
15 angepaßten, beschleunigten Betriebsart ausgeführt wird.

2. Verfahren nach Anspruch 1,
d a d u r c h g e k e n n z e i c h n e t,
daß die erste Abfolge von Verfahrensschritten eine taktzyklenbasierte Simulation des Mikrocontrollers/Mikroprozessors
20 und der Peripheriemodule vorsieht.

3. Verfahren nach Anspruch 1 oder 2,
d a d u r c h g e k e n n z e i c h n e t,
25 daß die erste Abfolge von Verfahrensschritten eine Folge von aufeinanderfolgenden Programmcodes ist.

4. Verfahren nach Anspruch 3,
d a d u r c h g e k e n n z e i c h n e t,
30 daß die Marken durch im Programmcode üblicherweise nicht verwendete Opcodes oder Opcodesequenzen gebildet sind.

5. Verfahren nach einem der Ansprüche 1 bis 4,
d a d u r c h g e k e n n z e i c h n e t,
35 daß während der zweiten Abfolge von Verfahrensschritten vorgegebene Peripheriemodule funktional mitsimuliert werden.

6. Vorrichtung zur Durchführung des Verfahrens nach einem der Ansprüche 1 bis 5,

g e k e n n z e i c h n e t d u r c h

- 5 eine Mikroprozessor-Steuereinheit zur Simulation des Moduls durch Erzeugung von im wesentlichen taktzyklengenauen Signalmustern sowie zur Abfrage und zur Auswertung der durch die Simulation hervorgerufenen Modulzustände während einer Programmunterbrechung durch Aktivierung eines Befehlssatz-
- 10 Simulators.

Zusammenfassung

Verfahren und Vorrichtung zur Systemsimulation von Mikrocontrollern/Mikroprozessoren und zugehörigen Peripheriemodulen

5

Es wird ein Verfahren zur Systemsimulation beschrieben, das sich auszeichnet durch eine erste Abfolge von Verfahrensschritten zur Simulation des Mikrocontrollers/ Mikroprozessors und der Peripheriemodule mit vorbestimmten Signalmustern und eine zweite Abfolge von Verfahrensschritten zur Abfrage und zur Auswertung von durch die Simulation hervorgerufenen Systemzuständen. Die erste Abfolge zur Durchführung der zweiten Abfolge wird nach Maßgabe von in die erste Abfolge eingefügten Marken unterbrochen und die zweite Abfolge in einer an die Auswertung angepaßten, beschleunigten Betriebsart ausgeführt.

10

15

Figur 1